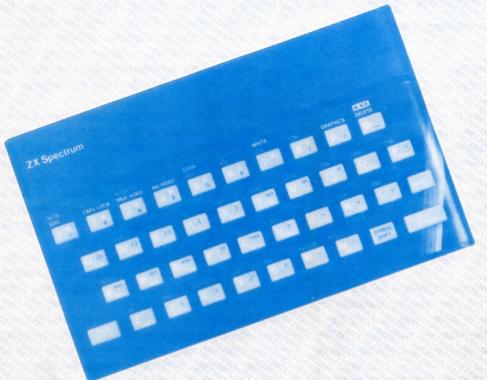
GLUBE



Julho/84

N.º 22

NESTE NÚMERO

INT. À LINGUAGEM MÁQUINA (Cont.) HISTÓRIA DOS MICROS SINCLAIR CONVERSÃO DE PROGRAMAS DO ZX81 (Cont.) OS PROCESSOS DA INTELIGÊNCIA ARTIFICIAL Programa ZX81/Spectrum
CONVERSÃO DE PROGRAMAS DO ZX81 (Cont.) 5 OS PROCESSOS DA INTELIGÊNCIA ARTIFICIAL 7
OS PROCESSOS DA INTELIGÊNCIA ARTIFICIAL
Programa ZX81/Spectrum
Master Mind 8
Dec-Hex 8
Evolução S
Vu-Meter II
Processador de Texto
Gráficos 3D
Cosmazoigs
Puzzle de Palavras
Rotina em Código Máquina (resposta à pergunta de
Mário Rebelo)
Duas Pequenas Rotinas
TOPS EM INGLATERRA 20
NOVOS PROGRAMAS

No Interior:

Cupão de Inscrição

Edição: Clube Z80

Fotocomposição: Fotomecânica Mabreu/Porto

Impressão: Ramos dos Santos & C.ª, Lda./Porto

Tiragem: 500 exemplares, Julho 1984

EDITORIAL

A revista CLUBE Z80, desde que se lançou, teve sempre como princípio fundamental não tratar a questão da Informática com uma concepção demasiadamente utópica e teórica, mas ser acessível, em termos práticos, ao utilizador mais inexperiente.

Nessa linha de ideias, julgamos estar a colocar realmente a máquina no seu devido lugar, articulando à prática a teoria necessária. Cremos que o conteúdo da revista cada vez mais se adequa às utilizações desejáveis e não a uma mera leitura que, depois, "fica na gaveta".

Contudo, parece-nos — e muitos sócios confirmam — que o esquema da revista está a tornar-se muito rotineiro. Há pouco tempo, Hugo Assumpção sugeria que variássemos um pouco, com novas abordagens, como p. ex., falar dos vários micros; Cibernética; Inteligencia Artificial; 5.ª Geração;

Vendas de Computadores; CLUBE Z80 — Curva de Crescimento, Materiais Vendidos, Balanço Anual, Colaboradores, etc..

Assim, vamos tentar responder a estas e outras pretensões que surjam, alterando um pouco a "parte técnica" da revista, pois pensamos que a parte de programas serve perfeitamente os desejos dos utilizadores.

Ficamos também à espera das suas sugestões. Elas serão anotadas na nossa pauta e procuraremos, na medida do possível, viabilizá-las.

ALEXANDRE SOUSA J. MAGALHĀES MARIA IRENE

INTRODUÇÃO À LINGUAGEM MÁQUINA

ZX81/SPECTRUM

Autor: FERNANDO PRECES

(Cont. dos números anteriores)

PARTE III - COMO FUNCIONA O Z80 (Cont.)

4 — Programação em código máquina

Depois de uma introdução superficial sobre os componentes internos dum microcomputador e alguns detalhes de funcionamento dos seus principais órgãos, estamos em condições de iniciar um estudo mais profundo que nos relacione com a linguagem de acesso directo ao microprocessador Z80.

As instruções em linguagem máquina aparecem, quer no Manual do ZX81, quer no Manual do ZX Spectrum, escritas em Assembler, que emprega mnemónicas alfanuméricas para mimbolizar operações e nomes dos vários órgãos do processador.

O Assembler é pois uma linguagem que utiliza as mnemónicas como idioma intermédio entre o Homem e a máquina.

Tal como as linhas de instruções BASIC, também o códido máquina é arrumado de igual forma na RAM, com a diferença que importa salientar do BASIC ter um endereço inicial invariável, e o C.M. poder ser colocado em qualquer endereço disponível da memória, que se encontre acima da RAMTOP. Porquê acima da RAMTOP? Pode interrogar-se o leitor menos esclarecido.

Porque abaixo dessa zona, as faixas de memória ocupadas pelas variáveis e pelas pilhas máquina (ver o manual) se deslocam a cada instante ao longo da RAM, conforme as solicitações da programação. O código máquina necessita, por razões que veremos mais tarde, de estar depositado em localizações permanentes, que serão obtidas depois de se interditar uma zona da memória à acção do monitor.

No ZX81, para de **consignar** a RAMTOP, uma vez determinado o endereço respectivo extensão, (este depende da extensão do bloco ou da rotina em C.M. que queremos introduzir), é necessário calcular os valores do (Low Byte Address) e do (High Byte Address), com os quais devemos carregar a **variável do sistema** (RAMTOP), endereços 16388/9.

Sendo X o endereço, teremos:

X = Y * 256 + Z e calcula-se:

M = (X/256)

Y = (High byte) = INT m

Z = (Low byte) = ((m - Y) * 256)

POKE 16 388, Z

POKE 16 389, Y e NEW

No Spectrum, é bastante mais simples.

Apenas:

CLEAR X e ENTER

Assim, um grupo de bytes acima da RAMTOP é totalmente ignorado pelas operações rotineiras da programação BASIC, e somente o comando POKE o pode alterar. NEW apenas limpa a memória abaixo da RAMTOP.

Para além de números, alfabeto e sinais ortográficos, o Basic do ZX81 possui uns 70 comandos utilizáveis a partir do teclado (o Spectrum tem uns 100), que podem desencadear através do monitor determinadas funções do Z80.

Dada a grande diferença existente entre as duas máquinas, já do conhecimento dos nossos leitores, teremos de admitir que essa, na sua maior parte, se situa a nível das ROMs (a do ZX81 com 8 K bytes e do Spectrum com 16 K bytes de extensão).

No ZX81, os endereços compreendidos entre 8.193 e 16.383, na realidade, não existem. A informação que recebemos com um PEEK n (endereços citados) dá-nos uma **imagem** da ROM, **repetindo os conteúdos** dos endereços entre 0 e 8.192, devido a um arranjo de ligações eléctricas, na base do chip.

Voltaremos mais tarde a falar sobre esse **espaço vago** e como o poderemos utilizar.

Para elaborar um programa monitor da envergadura destas ROMs, são necessárias algumas dezenas de técnicos altamente especializados nos campos das matemáticas e da electrónica digital. Outros requisitos não relacionados com a técnica também entram em linha de conta, pois é normalmente o fabricante do computador quem determina o que quer vender, mediante os produtos lançados no mercado pela concorrência, ou o tipo de utilizador que tem por objectivo atingir.

É esta última, a razão fundamental porque o Basic destas máquinas Sinclair é tão vigiado (elas vêm preparadas para o utilizador inexperiente). A única forma de contornar essa limitação, que torna o Basic-Sinclair extraordinariamente lento, é utilizar em linguagem máquina a alta velocidade do seu processador, aproveitando as potencialidades proporcionadas pela manipulação das 700 instruções do seu Assembler. Pegando nos manuais respectivos (ZX81, páginas 187 a 193 e Spectrum, páginas 171 a 178) podemos verificar que 252 dessas instruções são formadas por um único byte (todas as que se encontram na primeira coluna) e as restantes por 2 bytes.

Algumas dessas instruções exigem um complemento numérico que, associados, formam composições com 2, 3 e 4 bytes de extensão.

O tempo gasto na execução duma instrução Assembler é dependente dessa extensão, que está relacionada com o número de funções a cumprir, tais como:

- a) Tempo de comutação do apontador e recolha de uma cópia do conteúdo do endereço n. Esta acção é constante para todo o tipo de instrução.
- b) Câmbio de valores entre registros indigitados. Um grande número de instruções que envolvem um único byte e que provocam a carga de registro com conteúdo de outro, ou as que provocam o câmbio de conteúdos entre registros, são de execução muito rápida (soma dos tempos a e b).
- c) Recolha de cópia do conteúdo do endereço (n + seguintes) para o complemento numérico da instrução (1 ou 2 bytes). Este tipo de instrução é executada num tempo (a + c) ou (a + 2 c).
- d) Transferência dum conteúdo de registro para um endereço X da RAM. (Soma dos tempos (a e d) para 1 byte ou (a + 2 d) para 2 bytes).
- e) Intervenção da ALU (unidade de aritmética e lógica). O tempo de execução de algumas instruções é agravado por esta intervenção, que é muito curta pelo facto de esta unidade apenas efectuar pequenas operações binárias, mas que não pode deixar de ser referida.

A Unidade de Control uniformiza rigorosamente esses tempos, servindo-se de uma **batuta** (o ciclo máquina), que é sincronizado por impulsos introduzidos na entrada **Clock** do processador.

4.1 — O Clock e o Ciclo Máquina

Um circuito comandado a cristal de quartzo (tal como nos relógios electrónicos) fornece os impulsos de sincronismo que comandam toda a actividade do Z80. Este elabora, a partir deles, 2 ciclos máquina de diferente duração.

O primeiro (M1) contém 4 ciclos T (ciclo T é igual a um período do Clock) e o segundo (M2) contém 3 ciclos T.

Chama-se Ciclo de Instrução ao tempo de recolha e execução duma instrução que, como já vimos, pode ter uma maior ou menor duração, conforme a sua complexidade, sendo a extensão medida em ciclos T.

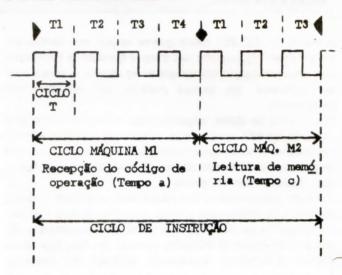


FIG. 4.1 — Exemplo de um Ciclo de Instrução para uma mnemónica que ordena a carga dum registro simples com um conteúdo do endereço (n + 1).

O ciclo T tem uma duração diferente em cada máquina (para o ZX81, 307,6 nanosegundos e para o Spectrum 285,7 nanosegundos), a que corresponde para o ZX81 uma frequência de trabalho de 3,25 megaciclos e para o Spectrum uma de 3,5 megaciclos.

No exemplo apresentado na figura 4.1, o tempo necessário para a execução dessa instrução é no ZX81 de 7 * 307,6 nanosegundos (2,1532 microsegundos → 2,1532 * 10⁻⁶ do segundo) e no Spectrum de 7 * 285,7 nanosegundos (1,9999 microsegundos → 1,9999 * 10⁻⁶ do segundo).

4.2 — O salto para uma rotina máquina

Tal como um GO SUB produz um salto para a Linha n, aonde se encontra a subrotina que se pretende executar e o RE-TURN provoca o regresso imediato à instrução seguinte do ponto de salto, também o comando USR X produz um salto para determinada rotina máquina que se inicia no endereço X e a mnemónica RET provoca o retorno incondicional ao Basic, à instrução seguinte do ponto de salto.

O comando USR, cujo argumento X é sempre um endereço, obriga o Z80 a **interromper** a sequência do programa monitor e a percorrer uma outra com **início em x** (1.º endereço da Rotina máquina) situada ou na RAM (C.M. do programador) ou na ROM (C.M. do monitor).

Note-se que num salto USR X, para uma rotina memorizada na RAM, **não podemos** utilizar o registro IY e o alternativo HL, sem primeiro memorizar os seus conteúdos, que terão de ser restituídos aos mesmos no instante de retorno (antes de RET).

Se tal não acontecer, ou o programa se imobiliza por estar desactivada uma interrupção de máscara (o apontador IY perdeu o seu endereço base), ou se dá um colapso total e tudo desaparece (o alternativo HL perde o ponto de reentrada no monitor).

Em qualquer dos casos, o utilizador terá de desligar a máquina e recomeçar o trabalho.

As formas mais usuais de se empregar nas 2 máquinas o comando USR, são:

PRINT USR X — Que executa o C.M. e regressa escrevendo no ecran o valor decimal contido no registo BC, na posição PRINT do ficheiro, endereçada pela variável DF CC (23 684/5, no Spectrum) e variável DF CC (16 398/9, no ZX81).

RANDOMIZE USR X ou RAND USR X — Que activa o gerador de números aleatórios antes do salto.

LET A = USR X — Que utiliza uma variável adicional, equivalente à função da rotina.

A mnemónica RET (código 201), obriga o Z80 a voltar de imediato à sequência interrompida quando do salto USR X.

Ensaio:

PROGRAMA 0 — Utilização do comando USR e da instrução RET.

ZX81

Para 1 K RAM RAMTOP em 17151, 2 K RAM em 18075, 16 K RAM em 32410.

Sugerimos que, para os ensaios a efectuar, o utilizador do ZX81, consoante a extensão de memória da sua máquina escolha para a RAMTOP, um dos endereços acima indicados.

mnemónicas

- 10 SLOW
- 20 POKE (RAMTOP + 1), 201

RET

- 30 STOP
- 40 LET A = USR (RAMTOP + 1)
- 50 PRINT AT 12,7; "DE VOLTA AO BASIC"

SPECTRUM

Para 16 KAM RAMTOP em 31999 e 48 K RAM em 64999

mnemónicas

- 10 CLEAR (RAMTOP)
- 20 POKE (RAMTOP + 1), 201

RET

- 30 STOP
 - 40 LET A = USR (RAMTOP + 1)
 - 50 PRINT AT 12,7; "DE VOLTA AO BASIC"

Depois de escrever o programa, → RUN. O único código desta Rotina vai ser memorizado e o programa pára na linha 30. A seguir → RUN 40 e o Z80 salta para o endereço indicado, lê o código de retorno e volta ao Basic (linha 50).

4.3 — As mnemónicas do Z80

GRUPO I

A instrução NOP (não operativo), nada mais faz de que provocar um tempo de atraso de 4 ciclos T na recolha da instrução seguinte, pois não obriga o processador a produzir qualquer trabalho. Nenhum registro ou flag é afectado.

A instrução é utilizada em C. M. como **enchimento** para produzir certos efeitos ou rectificar erros.

Exemplo:

Produzir um compasso de espera, para atrasar o movimento de uma figura, para a leitura duma legenda, apagamento de uma ou mais instruções que pretendamos eliminar, ou ainda,

para reservar espaços dentro duma rotina com o propósito de introduzir mais tarde algumas instruções que de início não tenham sido previstas.

mnemónicas	Código de operações	N.º Bytes	N.º ciclos M	N.º ciclos ·T			
NOP	0	1	1	4			

Ensaio:

PROGRAMA 1 — Instrução NOP. Para a RAMTOP, proceder como no ensaio anterior.

ZX81

mnemónicas

- 10 CLS
- 20 FOR A = 1 TO 99
- 30 POKE (RAMTOP + A), 0

- 40 NEXT A
- 50 POKE (RAMTOP + 100), 201

RET

NOP

- 60 STOP
- 70 LET A = USR (RAMTOP + 1)
- 80 PRINT AT 12,7; "DE VOLTA AO BASIC"

Para o Spectrum, substituir a linha 10 por CLEAR RAMTOP: CLS.

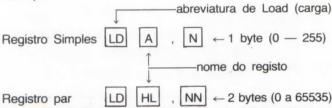
Introduzir RUN, aguarde o STOP e depois RUN 70.

Comentários:

A rotina é formada por 99 NOPs, e 1 RET.

GRUPO 2 — Este grupo é constituído por todas as instruções que representam a carga de registros com um ou dois bytes de dados, copiados dos endereços seguintes.

Exemplo:



Carga de registros simples

mnemónicas	códigos
LD A, N	62
LD H, N	38
LD L, N	46
LD B, N	6
LD C, N	14
LD D, N	22
LD E, N	30

Tempo de execução destas instruções:

N.º Bytes	N.º ciclos M	N.º ciclos T
2	2	7

Carga de registros pares

mnemónicas	códigos	
LD HL, NN	33)
LD BC, NN	1	a
LD DE, NN	17	1
LD IX , NN	221, 33	b
LD IY, NN	253, 33	
LD SP, NN	49	— a

Tempo de execução destas instruções:

Grupo	N.º Bytes	N.º ciclos M	N.º ciclos T
а	3	3	10
b	4	4	14

Neste grupo, os 3 primeiros registros pares da tabela são denominados **utilitários** e os 3 últimos **especializados.** Por enquanto não vamos abordar a utilização destes últimos.

Ensaio:

Programa 2 — carga de registros com dados memorizados nos endereços seguintes.

ZX81 e ((Spectrum))

mnemónicas

- 10 CLS ((CLEAR 64999))
- 20 LET A = 32412 ((CLS: LET A = 65000))
- 30 POKE A, 14
- 40 PRINT AT 5,0; "INTRODUZA UM VALOR (0 a 255)" "PARA O REGISTRO C"
- 50 INPUT C
- 60 POKE A + 1, C
- 70 POKE A + 2, 6

LD B, N

- 80 POKE A + 3, 0
- 90 POKE A + 4, 201

RET

- 100 PRINT AT 12, 0; "O CONTEÚDO DO REGISTRO C" ,," = "; USR 32412 ((USR 65000))
- 111 RUN ((RUN 20))

Nota: Para o Spectrum, escrever ou alterar na linha respectiva, o que estiver entre parentesis **dobrados**.

Comentários: como já fizemos referência, o comando USR, antecedido duma instrução PRINT, inscreve em decimal o conteúdo do Registro BC, no regresso ao BASIC.

Depois deste ensaio, altere a linha 70 para POKE A+2, 0 e corra o programa. Agora verifica-se um erro entre o valor introduzido no INPUT e o valor inscrito. Sabe explicar o que aconteceu?

Sim?... Claro, é isso mesmo...

Quando do salto para a rotina máquina, o registo BC recebe o endereço base dessa rotina. você apagou com o INPUT a parte "C" desse endereço, mas não apagou a parte "B" e ela regressa.

Programa 2 A - Idem

ZX81 e Spectrum

mnemónicas

- 10 CLS ((CLEAR 64999))
- 20 LET A = 32412 ((CLS: LET A = 65000))
- 30 POKE A, 1 LD BC, NN
- 40 PRINT AT 5,0; "INTRODUZA UM VALOR (0 a 65535)",,"PARA O REGISTRO BC"
- 50 INPUT BC
- 60 LET M = BC/256
- 61 LET Y = INT M
- 62 LET Z = ((M Y) * 256)
- 70 POKE A + 1, Z
- 80 POKE A + 2, Y
- 90 POKE A + 3, 201
- 100 PRINT AT 12,0; "O CONTEÚDO DO REGISTRO BC",," = "; USR 32412 ((USR 65000))
- 110 RUN ((RUN 20))

Comentários: 2 bytes (LOW DATA BYTE) e (HIGH DATA BYTE) encontrados no cálculo efectuado pelas linhas 60 a 62, vão carregar o Registro BC. De regresso ao Basic, esse número é inscrito em decimal.

Este grupo de instruções não afecta os flags.

(Cont. no próximo número)

HISTÓRIA DOS MICROS SINCLAIR

Trad. e Adapt. de "The Complete Sinclair Database"

PARTE I - 1962 - 1980

CLIVE MILES SINCLAIR (nascido em 1940), assistente editorial da revista «Practical Wireless», fundou em 1962 a firma «Sinclair Radionics» que comercializava pequenos transistores. Com a produção de novos equipamentos (rádio-transistores, micro-Tvs, calculadoras de bolso, relógios digitais, etc.), a companhia parecia expandir-se com muito êxito. Na realidade as coisas aconteceram ao contrário: deficiências nos equipamentos originaram o descrédito público na firma.

Sinclair mudou-se para a NEB (National Enterprise Board) criando aí o «design» de um computador chamado **NEW-BRAIN**, que acabou por vender à firma «Newbury» por achá-lo muito dispendioso.

Em 1978, sinclair produzia efectivamente o seu primeiro computador — o **MK 14.**

Contendo um processador SC/MP CMOS, com 1/4 K RAM, um teclado hexadecimal e um monitor com 1/2 K EPROM, o MK 14 foi colocado à venda por £: 43.55. O seu concorrente era um COMMODORE que custava o dobro.

Algumas deficiências, uma das quais era o teclado ser fabricado em borracha, originaram grandes críticas na imprensa e o MK 14 perdeu a popularidade que tinha conquistado.

Em Fevereiro de 1980, Sinclair lançou o **ZX80**, o primeiro computador em BASIC, reproduzindo um processador da NEC — o Z80A.

Por um preço inferior a £ 100, o ZX80 vendia-se em forma de «Kit» ou já completo, incluindo um manual de programação.

Possuindo 1 K de memória RAM, o ZX80 era a primeira oportunidade de qualquer pessoa possuir e utilizar um computador.

O BASIC ocupava apenas 4 K ROM e as instruções introduziam-se nos programas por simples pressão de uma tecla.

As encomendas atingiram um número tal, que a Sinclair teve que determinar prazos de espera de 3 meses que, mesmo assim, nunca eram cumpridos.

O mais importante no ZX80 era que, não obstante possuir apenas 1 K de memória RAM, tinha bastantes aplicações sem exigir «extras».

Pouco tempo depois vendia-se uma extensão, de memória de 3 K RAM e, mais tarde, 16 K RAM. Mas rapidamente a máquina começou a demonstrar os seus limites, dado que o microprocessador apenas realizava uma tarefa de cada vez.

Aproveitando o «design» anterior, logo se idealizou uma cópia — o MICRO ACE.

Com circuitos principais e BASIC ROM idênticos ao ZX80, embora possuindo 2 K RAM (e não 1 K), esta máquina veio a ser vendida, sob licença, aos E.U.A.

Melhoramentos ao «design» foram prometidos mas, na realidade, nunca vieram a público...

(Cont. no próximo número)

CONVERSÃO DE PROGRAMAS DO ZX81 →ZX SPECTRUM

Autor: FERNANDO PRECES

(Cont. dos números anteriores)

PARTE I

1.3 - PROGRAMA EM BASIC (Cont.)

Uma vez gravado o programa basic (v. número anterior), vamos introduzir na máquina o monitor que irá formar uma REM suportando 1751 bytes, necessários à introdução do código-máquina.

1 REM .* STEP \:PNOT ABS (.! MOVE \ STEP =?? DIM :QNOT SGN >= ? FOR XXXXXXXXXXXXXXXXX 20 LET a\$="042205092058080195189040014035034205092052046033209092205136015024233058080119518820024235" REM Primeira REM com qual quar extensao 30 FOR f=1 TO LEN a\$/3 40 POKE 23769+f, VAL a\$((3*f) -2 TO 3*f) 45 LET k=23769+f: PRINT k, PEEK k 50 NEXT f 51 STOP 55 INPUT "QUANTOS BYTES ? "; a 57 LET a=a+2 60 LET b=INT (a/256): LET c=(a/256-b)*256 60 RANDOMIZE USR 23770 1000 REM DEPOIS DE OBTIDA A RESERVA DE BYTES NECESSARIA AO CO DIGO MAQUINA, DEVE APAGGA TODAS AS RESTANTES LINHAS DO PROGRAMA CONFORME INSTRUCOES DO TEXTO.

Após ter escrito o monitor, corra o programa com RUN. No ecran surge a pergunta: «Quantos Bytes?»

Como resposta, introduza o número 1751, aguarde o aparecimento do relatório e LIST o programa.

Apague todas as linhas do basic, excepto a linha REM. Agora introduza:

> POKE 23755, 39 ENTER POKE 23756, 16 ENTER

Vamos chamar a seguir o programa basic «CONVERSOR» que se encontra gravado em fita, com:

MERGE «CONVERSOR»

Logo que surja o relatório, pode voltar a gravar o programa, agora já aumentado com a linha REM, com:

SAVE «CONVERSOR 1»

NOTA: VERIFIQUE o programa antes de o apagar da máquina, com:
VERIFY " "

Se tudo bateu certo, vamos continuar, com NEW e depois CLEAR 25984.

Seleccione a seu gosto um pequeno programa monitor que introduza código-máquina em decimal.

E agora vamos ao trabalho. Tem de escrever 1751 códigos, mas não o faça duma vez pois fica estoirado e vai cometer erros. Introduza 100 ou 200 códigos e depois pare. Pode gravar o que fez, apontando o último endereço e:

SAVE "0" code 25985, (o número de códigos escritos)

a segunda etapa, com:

SAVE "1" code (número apontado+1), (número de códigos escritos)

etc . .

1739517395173951739517395 999918833455567559999112395 999995655555555555555555555555555555	141435 371 2 434882994433 910 602414 49 4949498229 1501591442 49 494949498229	88 8	1 40 5879 49 325492535948	251150509140020553359457102 555091 7 344 2 8342205078 4	1594575000500406756886653501 2
--	---	------	---------------------------	--	-----------------------------------

5173951739517395173951739517395173951739	9001500600631301669116079464000560690719966419 901505669 9709944101635 04 9048054 95 4 590	21662662226221426133795421612311200100102061 07 67240016 545723363252600 90 90 1 4 7	16661331266222661792823261262462266122001010101121	50693307970 32034922 2264 540 35655 5 7 50693307970 32034922 2264 540 35655 5 7 7	50720222215222144255 5545579 430 55 5 542 2 50 0002175232144255 5545979 430 55 5 542	202190000000000000000000000000000000000
--	---	---	--	---	---	---

1739517395173951739517395173951739517395	9 61 7 5 60 0600 75 07 7 65 4444109553349717520279530 76 3 57 59444 57 59444	6 5 57 7 6 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9	3913956276662246611322662201411415339220000	1463311068726663136652266214617200721555000000	53355331494595152321921295729955953211999999	114153185265222666222257 5 7 144555 5 7 144555
--	--	--	---	--	--	--

(Cont. na próximo número)

OS PROCESSOS DA INTELIGÊNCIA ARTIFICIAL

O Professor António Costa, um brasileiro com profundos conhecimentos da Inteligência Artificial, fala dos princípais acontecimentos nessa área, ocorridos nos últimos dois anos:

- O Japão lança um projecto com 10 anos de duração, visando a construção de um computador da 5.ª geração. Ele será capaz de falar inglês e japonês, fazer traduções, programar outros computadores, observar cenas, etc.;
- 2. Os Estados Unidos entram na corrida para construir um computador de 5.ª geração, reservando verbas fabulosas para pesquisas em Inteligência Artificial só o Pentágono detém 1 bilhão de dólares para gastar em cinco anos. Não deixa de ser lamentável que parte das pesquisas em Inteligência Artificial seja financiada por instituições militares:
- Bancos de Dados capazes de entender Inglês, Japonês, Alemão, Francês e até Português começam a ser vendidos nos Estados Unidos. Entre os utilizadores de tais Bancos estão o Bank o America e a AVCO:

- A Xerox e a Symbolics Inc. construiram computadores especialmente projectados para executar programas em LISP, a principal Linguagem da Inteligência Artificial;
- Os japoneses anunciam para 1985 um computador pessoal capaz de executar eficazmente programas em PROLOG, a linguagem que ocupa o segundo lugar em popularidade entre as pessoas que trabalham com Inteligência Artificial.

CAMPANHA NOVOS SÓCIOS

Durante o mês de AGOSTO, o sócio que conseguir uma **nova inscrição** no CLUBÉ Z80 terá a oferta de uma CASSETE COM UM JOGO, «à sua escolha.

MASTER MIND

ZX81

Autor: FERNANDO PRECES

```
1>REM "F1J1"
5 GO SUB 450
10 CLS
15 DIM C(4)
20 DIM G(4)
30 LET C(1) = INT (RND*9) + 1
40 FOR C(X) = INT (RND*9) + 1
60 FOR C(X) = INT (RND*9) + 1
60 FOR C(X) = INT (RND*9) + 1
60 NEXT J
90 NEXT J
100 FOR G(10 THEN GO TO 40
100 FOR G(10 THEN);">
100 FOR G(10 THEN GO TO 40
100 FOR G(10 THEN);">
100 FOR G(10
```

```
350 IF W<>1 THEN PRINT "S";
360 PRINT THEN PRINT TAB (5);
370 IF B=4 THEN PRINT TAB (5);
370 IF B=4 THEN PRINT TAB (5);
380 IF B=4 THEN PRINT TAB (6);
380 IF B=4 THEN PRINT TAB (6);
400 IF B=4 THEN PRINT TAB (6);
400 IF B=4 THEN PRINT TAB (6);
410 INPUT L$
410 INPU
```

D]=(0<u>H</u> H=);(0

TS 1000/ZX81/SPECTRUM

Autor: CARLOS MORENO

"Este programa, tal como apresenta a listagem, funciona apenas no ZX81 ou no TS 1000. Apesar disso, podemos pô-lo a funcionar no ZX SPECTRUM com as seguintes alterações:

Linha 12 será: 12 LET N = 23760 Linha 61 será: 61 LET N = 23760

Outro modo possível de alterar este programa para o SPECTRUM é:

Linha 10 será: 10 CLEAR 5000 Linha 12 será: 12 LET N = 50001 Linha 61 será: 61 LET N = 50001

Ao ser executado, o programa passa os números introduzidos de hexadecimal para decimal, colocando-os em seguida na instrução REM. Para o microprocessador poder executar esta tarefa, devemos fazer RAND USR 16514. Atenção! . . . se for introduzido um Código Hexadecimal maior que FF, será produzida uma mensagem de erro e o programa pára a sua execução.

```
10 REM ...
   12
       LET N=16514
LET L1=1
PRINT "CODIGO HEX EM ";N+L1
   25
-1;"
30
31
        INPUT
                 B$
       LET
              P=1
   32345
              R=0
                       O 1 STEP
        FOR
              L=LEN
K=L T
                     TO
        LET
               1=0
   36
            CODE B$(K) >37 THEN LET I
   37
38
39
        LET
              A=(CODE (B$(K))-28-J) *P
       LET
LET
NEXT
CLS
             P=P*16
R=R+A
   40
   47
       IF B$="" THEN GOTO 62
POKE N+L1-1,R
LET L1=L1+1
GOTO 20
   48
  50
              N=16514
NT "PREPARE O GRAVADOR"
   61
        LET N
        PAUSE 4E4
   63
   64
       CLS
       CLS
SAVE "CARREGADOR".
FOR F=N TO N+L1-1
PRINT F,PEEK F
NEXT F
  65
   70
```

EVOLUÇÃO

SPECTRUM 16/48 K

In. THE BEST OF SINCLAIR PROGRAMS/84
Trad.: J. MAGALHÄES

Aqui está um programa para quem gostar de Biologia, mais concretamente sobre evolução, selecção natural.

Em determinada população de ratos distinguem-se uns de côr preta, outros amarelos e, por «selecção», alguns serão eliminados enquanto outros poderão sobreviver.

A selecção não segue qualquer fórmula pois iria afastar-se bastante da realidade. Assim, será determinada ao acaso pela instrução RANDOM, mas dependente da percentagem que pretender.

1 DIM as (250): DIM b(51)
10 GO SUB 9000
15 BORDER 7: PAPER 7: CLS
20 PRINT INK 1; Este programa
simula determinadapopulação d de ratos, onde seDistinguem doi s tipos: "'ratos pretos # "'ra tos amarelos"; INK 6; "#"' INK 2; "Esta diferenca e originada p orum gene com dois alelos." "Y (preto) e"'"dominante sobre y (am arelo)." amente ao meio)"
29 PRINT #0; INVERSE 1;"Qualqu er tecla para continuar": PAUSE 30 CLS : PRINT INK 2; "Podes ntar qualquer uma destassituacoe s, e se ocorrer seleccaodetermin ar o seu numero"' INK Ø; INVERSE 1; "Qualquer tecla para comecar 35 IF INKEYS="" THEN GO TO 35
40 BORDER 7: PAPER 7: CLS
50 PRINT AT 7,0; INK 2; "Popula
Cao maxima?" "(Nao superior a 12
5)" 52 INPUT PØ: IF PØ<1 OR PØ>125
THEN PRINT INK Ø; "Deve ser en
tre Ø a 125.": GO TO 52
55 CLS: PRINT RT 7,0; INK 1;"
A seleccao pode ser: "' "Nao favo
ravel aos amarelos- "Ø" "" "Nao
favoravel aos pretos- "" "" ""
Nao ha seleccao-

57 INPUT d: IF d > 0 AND d < > 1 A ND d < > 2 THEN PRINT INK 0; ''Deve ser 0,1 ou 2": GO TO 57 58 IF d=2 THEN GO TO 75
60 CLS: PRINT AT 7,0; INK 2; "
POPULACIO SELECCIONAS '"(em percentagem)' "100% significa que
toda a populacao nao sobre
eviveria ""De entrada de um nu
mero ate 100" IF sp 0 OR sp 100
THEN PRINT IGO TO 62
THEN PRINT IGO TO 62
TO LET S=100-sp
75 CLS: PRINT AT 7,0; INK 1; "
Qual a percentagem inicial
de atelos amarelos ? (9%) " "0
THEN PRINT INK 0 '"0
TO NPUT Y: IF y 0 OR y 100 TH
EN PRINT INK 0 '"0
ET PP0 (1) = y
20 LET Sb=1: LET Sy=1: IF
110 LET y=y 100 ET Sb=5/100
215 BORDER 7: PAPER 7: CLS
220 LET Sb=1: THEN LET Sb=5/100
265 IF n=51 THEN PAUSE 120: GO
TO 310 E 2369 PRINT #0; BRINTUAR
265 GO SUB 5000
310 GO 310 S 5000
311 GO 500 SUB 5000
311 GF is="m" OR is="M" THEN GO
310 GO 300 SUB 5000
311 GF is="m" OR is="M" THEN GO
310 GO 300 SUB 5000
311 GF is="m" OR is="M" THEN GO
311 GF is="m" OR is="M" THEN GO TO 285 SUB 5000 SUB 5000 i \$="m" 0 310 GO GO 311 315 IF TO 215 320 IF TO 40 OR is="M" THEN GO i\$="n" OR i\$="N" THEN GO TO 40
325 IF i\$="s" OR i\$="S" THEN PA
PER 7: STOP
350 GO TO 311
1000 PRINT AT 0,7; INK 2; BRIGHT
1; "PROXIMA GERACAO ("; N-1;")"
1005 LET ym=0: LET cd=0: FOR f=1
TO 2*P 1010 LET x=RND: LET v=RND 1020 LET z=(x>=y)+(v>=y) 1030 LET cd=cd+z 1040 LET as(f)=STRs (z) 1045 LET i=0: IF as(f)="0 1020 LET z=(x>=y)+(v>=y)
1030 LET cd=cd+z
1040 LET as(f)=STR\$ (z)
1045 LET i=0: IF a\$(f)="0" THEN
LET i=6: LET ym=ym+1
1046 PRINT INK i;"2";
1050 LET y=(4*P-cd)/(4*P)
1065 PRINT AT 19,0; INK 2; BRIGH
T 1;ym;" ratos amarelos em ";
2*P)'(4*P-cd);" alelos amarelos
em ";(4*P)
1070 PRINT AT 19,0; INT AT 0,2; INT
2*P)'(4*P-cd);" alelos amarelos
em ";(4*P)
1070 PRINT AT 1;" NEM YOU
2000 PRINT AT 0,2; IND
2000 PRINT AT 0,2; INT
2000 PRINT A 2072 IF ((a\$(f)="0") AND (x)f*sy
)) THEN LET ym1=ym1-1: LET a\$(f)
="d": LET ps=ps-1: PRINT "
2075 IF a\$(f)
1: PRINT OV
ER 1; INK 8;"
2080 NEXT f
2082 PRINT AT 20,31;"
2083 PRINT AT 0,0;"
2084 IF bm = 0 THEN PRINT AT 19,0;
BRIGHT 1; INK 2; bm1; "PRETOS 50
BREV. DOS "; bm; INK 1; TAB (28); INT (100*bm1/bm+.5); "%"
2086 IF ym>0 THEN PRINT AT 20,0;
BRIGHT 1; INK 2; ym1; "AMARELOS
SOBREV. DOS "; bm; INK 1; TAB (28)
; INT (100*bm1/bm+.5); "%"
2090 LET ym1/ym; INK 1; "AMARELOS
SOBREV. DOS "; ym; INK 1; "AMARELOS
SOBREV. DOS "; ym; INK 1; "AMARELOS
SOBREV. DOS "; ym; INK 1; "%"
2090 LET (2*Ps-cd)/(2*Ps): PRI
NT BRIGHT 1; INK 1; "%y="; (INT (y
*1000+.5))/10;" (Ultima geraca
0 = "; (INT (b(n-1)*1000+.5))/10;"
2095 LET b(n)=y
2098 LET p=p0: IF ps (=p0 THEN LE
T p=ps
2100 RETURN
5000 IF INKEY\$="" THEN GO TO 500
5020 LET i\$=INKEY\$: RETURN
6000 BORDER 1: PAPER 1: CLS : IN
K 7: PLOT 47,156: DRAW 0,-120: D
RAW 200,0
6010 FOR f=36 TO 156 STEP 24: PL
OT 45, f: DRAW -3, 0: NEXT f
6020 FOR f=47 TO 247 STEP 40: PL
OT 45, f: DRAW -3, 0: NEXT f
6020 FOR f=47 TO 247 STEP 40: PL
OT 45, f: DRAW -3, 0: NEXT f
6020 FOR f=47 TO 247 STEP 40: PL
OT 45, f: DRAW -3, 0: NEXT f
6020 FOR f=47 TO 247 STEP 40: PL
OT 6,36: DRAW 0,-4: NEXT f
6020 FOR f=47 TO 247 STEP 40: PL
OT 6,36: DRAW 0,-4: NEXT f
6020 FOR f=47 TO 247 STEP 40: PL
OT 6,36: DRAW 0,-4: NEXT f
6020 PRINT AT 9,0; "%y"; AT 2,2;"1
6030 PRINT AT 9,0; "%y"; AT 2,2;"1
600"; AT 5,3; "80"; AT 8,3; "60"; AT 1

1,3;"40";AT 14,3;"20";AT 17,4;"0
6040 PRINT AT 18,6;"0 10 20
30 40 50";AT 19,15;"Geraca
6050 PRINT AT 1,8; INVERSE 1;"Po
pulacao max. ";p0
6060 If d=2 THEN PRINT AT 0,2; I
NVERSE 1;"Nao houve seleccao "
6070 If d=1 THEN PRINT AT 0,2; I
NVERSE 1;INT (sp);"% nao a favor
dos pretos"
6080 If d=2 THEN PRINT AT 0,1; I
NVERSE 1;INT (sp);"% nao a favor
dos amarelos"
6080 If d=0 THEN PRINT AT 0,1; I
NVERSE 1;INT (sp);"% nao a favor
6080 If d=0 THEN PRINT AT 0,1; I
NVERSE 1;INT (sp);"% nao a favor
6100 PLOT 46,120*b(1)+36: DRAW D
DER 1;3.0
6110 FOR f=2 TO n: PLOT OUER 1;4
2+4*f,120*b(f)+36: NEXT f
6115 IF n=51 THEN GO TO 6200
6120 PRINT AT 21,0; INVERSE 1;"T
ecta ""M"" CONTINUAR": PRINT
Ecta ""M"" CONTINUAR": PRINT
Ecta ""M"" CONTINUAR": PRINT
Ecta ""M"" CONTINUAR
""S"" - PARR
"1;"Grafico completo" FLASH
0;"M- continua mas limpa o graf
1;co" "(n- comecar de novo, s- pa
1;co" "(n- comecar

VU-METER II

SPECTRUM 16 K

In. YOUR COMPUTER, Dezembro/83 Trad.: J. MAGALHĀES

VU-METER II

No n.º de Dezembro/83 do CLUBE Z80, publicamos o programa VU-METER (corrigido) que nos dava a representação gráfica de sinais audio.

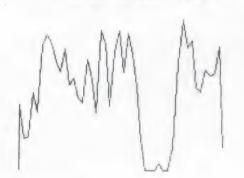
Devido ao entusiasmo demonstrado por vários sócios, decidimos publicar este novo programa que funciona de forma idêntica ao anterior, mas com pequenas alterações.

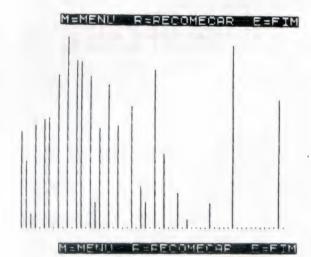
Temos agora 3 tipos de representações gráficas:

- Normal
- Barras
- Pontos

que supomos tornarem mais fácil a observação das diferentes tonalidades.

M=MENU R=RECOMECAR E=FIM





102 DEF FN A(L) =1+INT (.5+L/30)
103 GO SUB 405
105 GO TO 200
110 DRAW INK FN A(L); X-N,-L+Y:
LET X=N: LET Y=L: RETURN
120 DRAW INK FN A(L); 0,-L: RETU
RN
200 PAPER 0: INK 7: BRIGHT 1: C
LS
210 PRINT AT 1,9; INVERSE 1; "SC
ANNER DE SOM"
215 PRINT AT 3,1; "Este programa
faz a representacao grafica de
sinais audio pela entrada do EAR
do Spectrum"
220 PRINT AT 8,1; "Introduza uma
cassete (ex: musica), inicie o
gravador e seleccione uma das se
guintes opcoes: "
225 PRINT AT 13,8; "1) NORMAL"; A
T 15,8; "2) BARRAS"; AT 17,8; "3) P
ONTOS"
230 INPUT INVERSE 1; "QUal a opc
a0 (1-3)"; Q: IF 0<1 OR 0>3 THEN

250 CLS : PRINT AT 0,7; INVERSE 1; "SPACE PARA PARA": LET I\$=""

LET X=0: LET Y=0

255 FOR N=0 TO 255 STEP 4: LET L=USR TONE: PLOT N,L

260 IF 0<3 THEN GO SUB 100+(0*1

265 LET I\$=INKEY\$: IF I\$=" "TH EN GO TO 300

270 NEXT N

275 GO TO 250

300 PRINT AT 0,5; INVERSE 1; "M= MENU R=RECOMECAR E=FIM": PAUSE 0

305 LET I\$=INKEY\$: IF I\$="M" TH EN RUN

310 IF I\$="R" THEN GO TO 270

315 IF I\$="R" THEN STOP 320 GO TO 300

400 DATA 1,0,255,17,0,0,219,254

203,119,32,1,19,16,247,66,75,20

405 LET TONE=32600

410 FOR N=TONE TO TONE+17: READ D: POKE N,D: NEXT N: RETURN

PROCESSADOR DE TEXTO

SPECTRUM 16/48 K

In. YOUR COMPUTER, Março 1983 Adapt.: J. MAGALHÃES

* Este texto foi elaborado com o programa que a seguir passamos a descrever.
Hetirado da revista Your Computer, Marco
de 1983.
For falta de disponibilidade, o programa
a apresentado com todas as instrucoes em
Ingles, no entanto supomos que a nossa
descricao e bastante elucidativa, facilicando a sua utilizacao.

Como ja teve oportunidade de verificar, este "processador de texto", permite ultra passar o limite normal de caracteres do opectrum (32 - 42), dando-lhe ainda a posbilidade de uma nova redefinicao. Neste Caso estao a ser usados caracteres de seis por oito pixels.

O programa inicialmente apresenta um menu de seis optoes:

— a primeira optao, da entrada para a cricao do texto, ou verificacao do mesmo.

Unicialmente e-lhe pedido o n. da pagina pretendida que sera apresentada seguidamente com o cursor posicionado no canto superior esquerdo do ecran, podendo, com as teclas 5 à 8, move-lo para qualquer posicão. Se acontecer sobreposição de caracteres, ficarão na memoria apenas os ultimos a ser transcritos.

São possiveis todos os caracteres, para o qual pode utilizar as teclas: Caps shift e symbol shift. Em caso de erro*na entrada de um caracter, para o apagar basta usar, como normalmente as teclas Caps shift e belete.

Se pretender iniciar uma nova linha, faca entre.

Se pretender iniciar uma nova linha, faca Enter, no entanto note que se estiver na Utima linha, nao avancara para a proxima Pagina, ficando o cursor dividido em duas Partes.

Verifique que no fundo do ecran se encontram ainda mais tres opcoes: Next- Passara a pag. seguinte, (sao apenas

Utilizadas 4 pags).
henu- Para regressar ao Menu.
Lopy- Fazer a copia do texto.(tecla Z).
Estas opcoes sao conseguidas accionando
simultaneamente as teclas: Caps Shift e
Symbol shift, seguindo-se a entrada da sua
inicial (inverse video).

taps shift + tecta 1), do quat constam as seguintes opcoes:

Insert - Para introducao de caracteres nao introduzidos por engano ou erro. (Tecta I) Posicione o cursor no local indicado (para correccao) e pelo modo Edit, accione a tecta "I". Introduza o numero de caracteres a entrar, num limite de 0 a 255. Pode fazer esta entrada de duas formas: ex. 1 ou w21. Neste ultimo exemplo a instrucao Enere e reita automaticamente.

Delete - esta opcao e identica a anterior sendo utilizada para retirar caracteres na linha, onde ocorreu o erro. E pretendido tambem o numero de caracteres a sair, sendo esta operacao feita pelo mesmo modo da opcao anterior.

trase - Limpa determinada parte do texto, desde a ultimo caracter a linha imediatamente abaixo do posicionamento do cursor, quando requerida esta opcao.

Justify L/r - Funciona para acerto de uma linha ou de toda a pag. (cursor line ou page, respectivamente). Experimente num pequeno texto deixar dois espacos antes de iniciar uma linha. Accione a tecta "l" e indique se pretende a correção nessa linha ou em todo o texto (c/p). Coloque, antes desta opcao o cursor no inicio do texto, se optou por; page (p), ou na linha se optou por linha (c). Verifique que o texto ou linha, ficaram juntas a margem esquerda pela opcao "r" ldireita), conseguira o acerto de linhas entre as duas margens do ecran, pela mesma forma da opcao anterior. "Todo o texto ou uma linha).

voltando ao Menu principal tem ainda as upcoes:
Stop- Parar o programa. (Com a instrucao continue" regressara ao menu). Em caso de brake ao programa, faca Goto 9222 e nao hun.

trase a page- Apagara toda a pag. indicada :1-4, save e Load- Funcionam da mesma forma de gravacao e carregamento de um programa normal, atribuindo um nome que nao deve exceder os 10 caracteres.

unange typeface- Contem tres opcoes:
Normal, New e Redefine.
Normal, tera os caracteres normais do Spec

Maiusculas).

New, para utilizar os caracteres previamente definidos, pela opcao Redefine.

Redefine— Para redefinir qualquer, de entrada do caracter, seguindo—se o codigo decimal correspondente. (Ver tabela de Conversao — Jornal n. 22 Clube 232).

Resim que o caracter vai sendo definido, podera acompanhar a sua formacao pelo dis—play apresentado numa "grelha" de 1—3 li—nhas com colunas de a—f. Se pretender ficar com o quadro dos novos caracteres deve gravar o programa com o comando QUTO 9222.

Como carregar o programa ?

Carregue o programa em BASIC (list. 1) e grave-o com o comando: GUTO 9900; no final da gravacao o programa fara "auto-run", ricando no modo LUPO; faca BREAK e verique a gravacao (UERIPY "").

Para introducao do codigo maquina, Utilize um "carregador" a sua escolha, nao esquecendo fazer CLEPR 29665, antes de iniciar esta operacao.

"Utilize por exemplo, o carregador do c.m., publicado no boletim n. 12 do CLIBE 290, pag. 14 "FIRE POX").

Para gravacão do cod. maq. (depois de introduzido completamente), faca:
SAUE "spc"CODE 29665,3122; verifique.

Note: U programa Basic deve estar gravado na cassete antes do cod. maq.

Lepois de completas todas as operacoas, com o cod. maq. intruduzido sem erros, com o programa gravado devidamente, pode verificar o seu trabalho.

Inicie o gravador com a cassete no principio do programa e faca: LOPD "", o cod. maq. entrara automaticamente.

U programa comeca pela apresentacão do Menu principal. BOM TRABRILHO!

Listagem 1

1000 RANDOMIZE USR 29900: GO TO RANDOMIZE USR 30434: STOP 5000 GO TO M RANDOMIZE USR 29915 IF PEEK V>CODE "4" THEN GO 2010 3020 3050 RANDOMIZE USR 30545: GO TO 4000 RANDOMIZE USR 30615: INPUT 18 4010 PRINT AT 11,x;"SAUING:""";i \$;""" DATA x*()": SAUE is DATA x *(): GO TO 5020 5000 RANDOMIZE USR 30674: INPUT \$(): GO TO 5020
5000 RANDOMIZE USR 30574: INPUT
1\$10 RANDOMIZE USR 30720: PAUSE
X: LOAD is DATA X*()
5020 RANDOMIZE USR 30755: PAUSE
X: LOAD is DATA X*()
5020 RANDOMIZE USR 30755: PAUSE
X: GO RANDOMIZE USR 29927
THEN RAN
5000 IF PEEK V=CO: GO TO MHEN RAN
5000 IF PEEK V=CO: GO TO MHEN
50050 IF PEEK X=CO: GO TO MHEN
50050 IF LEN is>1 OR is ("OR is
50050 IF LEN is>1 OR is ("OR is
50050 IF LEN is>1 OR is ("OR is
50050 IF LEN is>1 OR is
50050 IF LEN is>1 OR is
50050 IF LEN is>1 OR is
50090 RANDOMIZE USR 30157
5140 RANDOMIZE USR 31063: INPUT
510200 RANDOMIZE USR 31063: INPUT
510200 RANDOMIZE USR 31063: INPUT
510200 RANDOMIZE USR 300521: PAUSE
50020 RANDOMIZE USR 300521: PAUSE INKEY: PRINT #x; "Mage or MUTSO!
Line ?"
8020 RANDOMIZE USR 30021: PAUSE
x: IF PEEK v=CODE "C" THEN RANDO
MIZE USR 31214: GO TO t
8030 IF PEEK v=CODE "P" THEN RAN
DOMIZE USR 31225: GO TO t
8040 GO TO 760
9000 RANDOMIZE USR 29939: GO TO
183*(PEEK v-40)
9800 CLEAR 29665: LOAD ""CODE 29
666: GO TO 10
9900 SAUE "type" LINE 9800: SAUE
""CODE 29665,3102

Listagem 2

2573246632211367153825787615111511232212424223221 9500000 \$\text{\$\ 190039122137311232905111111322211113121254223231 32 214 253 573232 925323225 522 2534 3552450 573232 9237 555 565 7427 444 7 64 3668 6 11066001300385061506 8 636 64 566 688 244 685 884 868 488 568 648 110010500010001000000 0 6 6 6 620 240 240 86 0 6 6 620 240 240 86 4255 054 555 524 45 55 2 54 5 555 215599662965191759859966994799489965441699 010046051605664428 2 4 45 254 554 54 55 442 55 35 3544 2352000550 50555061305509115005591550 32522 32528 32534 32546 32546 32558 03610564 5164 542 5650 5650 612 426 8668 248 882 8 1224 070649 0624946 408 956646 44795 97 594 54 7 1 111222112121233224241242233331 956 18796869 68912699 628 500000000 5543 5543 5555 5524 5553 5153 1553 4462 0564 552 25 05654 552 25 228 548 952 5 1369185951391499 475524 5566 5566 4524 68 4 0000000 40 10 40 15 234

GRÁFICOS 3D

SPECTRUM 16/48 K

Autor.: Manuel Quinaz Porto

Após a entrada do programa, ser-lhe-á pedido Z = função (x,y), seguindo-se os valores iniciais e finais para x e y. É então traçado o gráfico em 3 dimensões e em perspectiva dessa função.

SEN X

X varia entre -8 e 8 Y varia entre 0 e 1

EXEMPLOS:

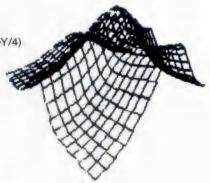
SEN X + COS Y X varia entre 0 e 8 Y varia entre 0 e 8





 $\mathsf{EXP}\ (-\mathsf{X}\!*\!\mathsf{X}) + \mathsf{EXP}\ (-\mathsf{Y}\!*\!\mathsf{Y}/4)$

X varia entre -4 e 4 Y varia entre -4 e 4



1/(ABS(x*Y))* . 5

X varia entre -3 e 3

Y varia entre -3 e 3

Jose Quinaz Manuel Em 6/7/1 RAVADOR ": DIM X (21,21): FO TO 20: FOR m=0 TO 20: LET (n+1) = (45568-850*m+5202*n)/ FOR n 7*(m+n)): NEXT 9910 0 CL5 : PRINT permite tridimensional 130 IF xi=xf THEN PRIN "ERRO: O valor inicial valor final.": G 140 INPUT "Valor inici gi: INPUT "Valor final 150 IF yi suf THEN PRI "ERRO: O valor inicial valor final.": (170 POKE 23674,0: POKE POKE 23672,0 160 LET m=1: LET n=1: THEN PRINT FLASH 190al TO 140 23673,8 POKE Zi=FN f(): LET Z X = Xi TO X STEP 41 STEP

(-ui)/20.9999 200 If m=2 AND n=1 THEN GO SUB 9800 205 LET z(m.n)-*** 205 LET z(m,n)=FN f()*(x(m,n)+5)/(178+17*(m+n-2))
210 IF z(m,n)>zf THEN LET zf=z(IF Z(m,n) (Zi THEN LET Zi=Z(m,n)
230 LET n=n+1: NEXT y: LET n=1:
LET m=m+1: NEXT x
240 LET a=55/(zf-zi)
300 CLS: FOR m=1 TO 21: FOR n=
1 TO 21
310 LET x=x(m,n): LET y=176*(17
*m-126+x(m,n))/(176+17*m)+a*(z(m,n)-zi): PLOT x,y: IF n>1 THEN D
RAW x(m,n-1)-x,176*(17*m-126+x(m,n-1))/(176+17*m)+a*(z(m,n-1)-zi)
-y: PLOT x,y
320 IF m>1 THEN DRAW x(m-1.0)-y
320 IF m>1 THEN DRAW x(m-1.0)-y)-y: PLOT x,y
320 IF m>1 THEN DRAW x(m-1,n)-1
.178*(17*(m-1)-128+x(m-1,n))/(1
8+17*(m-1))+a*(z(m-1,n)-zi)-y
340 NEXT n: NEXT m
400 INPUT "Parar 7: (s/n) "; r*
410 IF r*="s" OR r*="5" THEN 5 400 410 THEN ST IF (\$<>"n" AND (\$<>"N" THEN TO 400 INPUT "Nova funcao?: (\$/n) "S" THEN GO OR CER"N" THEN GO 120 GO 450 GO TO 430 810 LET \$20 *INT (.4*PEEK 23572+ .02.4*PEEK 23573+25214.4*PEEK 23 674-5)
9820 LET min=INT (seg/60): LE
eg=seg-60+min: LET hor=INT (m
60): LET min=min-60+hor
9830 PRINT RT 0,2;"Demorara a
ximadamente"
9840 PRINT RT 2,2;hor;" Horas MAC PRING MAC PR 9850 RETUI 9910 FOR 1 TO 30: BEEP .01, n

COSMAZOIGS

SPECTRUM 16/48

Baseado no jogo Asteróides, apenas com algumas alterações gráficas.

O JOGO:

O Objectivo do jogo é destruir todos os objectos que possam colidir com a nave que controlas. Podes rodar a nave, para qualquer direcção (teclas 6 e 7) ou mesmo perseguir os cosmazoigs (tecla 9), para disparar (tecla 0) e a (tecla 8) para parar o jogo.

Para esta tua missão tens apenas 3 naves, por isso escolhe o nível de jogo que preferes, (de 0 a 9).

Se conseguires o record, escreve o teu nome e a seguir "ENTER", caso contrário, regressarás à 1.ª letra.

O PROGRAMA:

Retirado da revista "YOUR COMPUTER", Mar/84.

É necessário introduzir 2 programas para conseguires o jogo:

 (listagem 1) que deves gravar com o comando: SAVE "Cosmazoigs" LINE 1 e verificar depois de gravado (VERIFY""). 2 — (listagem 2) introduz o código máquina com o máximo de atenção para evitares erros. Depois de passado, para verificares se tudo está em condições, far "RUN". Se ocorreu algum erro, o computador indicar-te-à a sua localização.

Se tudo está perfeito, obterás a informação: "PODES GRAVAR". Em último caso, para verificares se realmente tudo está correcto, podes fazer:

LET L = USR 30448 : BORDER 0

Se houver incorrecção o programa "Aborta".

Listagem 1

20 BORDER 9: PAPER 9: INK 7
30 BRIGHT 1: CLEAR 30440
35 POKE 23676,127
440 PRINT AT 11,4; FLASH 1; """C
05A20IGS""A CALCEGA!"
50 PRINT AT 9,10; INVERSE 1; F
LASH 1; "espece"
70 FOR F=12 TO 15: BEEP .05,F
80 NEXT F
90 INK 0: PRINT AT 15,0;
92 LOAD ""CODE
95 INK 7: PAPER 0: CLS

96 PRINT AT 10,11; "COSMAZOIGS" 97 GO SUB 2300 98 DIM NE(11,22) 99 FOR F=1 TO 11 105 LET NE(F) = "00000 ": NEXT F

107 CLS
110 PRINT AT 6,11; "OS PONTOS:"
120 PRINT AT 6,6; INK 2; CHR 14
5; INK 7; "... 100 PONTOS:"
130 PRINT AT 10,8; INK 4; CHR 14
46; INK 7; "... 100 PONTOS:"
140 PRINT AT 12,8; INK 6; CHR 14
47; INK 7; "... 1000 PONTOS:"
145 PAUSE 100: CLS
150 PRINT AT 4,9; "USE AS TECLAS 160 PRINT AT 7,8;"6> Rodar esquerda" PRINT AT 9,8;"7> Rodar dire 180 PRINT AT 11,8;"6> Parar 0 j 180 PRINT AT 11,8;"8 > Parar 0 j
090"
190 PRINT AT 13,8;"9 > Avancar"
200 PRINT AT 15,8;"9 > F090"
210 PAUSE 100: CLS
220 PRINT TAB 11; "RECORDS: "
230 PRINT TAB 5; INK 6-INT ((F
240 FOR F=1 TO 10
245 PRINT 'TAB 5; INK 7; N\$(F, 5
TO)
250 NEXT F
260 PAUSE 100: CLS
270 IF INKEY\$()" THEN GO TO 26
5 520 FOR G=1 TO 21. 540 NEXT G 545 FOR G=232322 TO 23295 545 POKE G,0: NEXT G 550 PRINT AT 7,5; INK 4; "0 methor of de hoje"; AT 9,1; "score maximo escreva o seu"; AT 11,14; "Nome"; AT 14,5; INK 6-INT ((F-1)/2); x 8; INK 7; G-11; GOPT GATE INK 7;"
560 LET G=11: POKE 23656,0
570 LET A: INKEY:
560 IF A: "THEN GO TO 570
590 IF CODE A: 13 THEN GO TO GO TO 570 THEN GO TO 68 0 500 IF R8 (" " OR A\$ > "Z" THEN GO TO 570 510 PRINT RT 14,G; R8

520 LET N\$(F,G-4) =A\$
530 BEEP .01,20: LET G=g+1
540 IF G=27 THEN LET G=11
650 IF INKEY\$()"" THEN GO TO 65 550 GO TO 570 570 PRINT AT 11,11; FLASH 1; "AC 500 FOR F=1 TO 100: NEXT F 690 GO TO 107 1010 DATA 5,5,6,1.25,1,.75,8,6.5 1020 DATA 8,1,5,1.5,1.5,1.5,5.7,8,7 1020 DATA 8,1,5,1.5,1.5,1.5,5.7,8,7 1030 DATA 1,.5,6,T,5,T,1,T,1,2.5,9 1030 DATA 1,.5,6,T,5,T,10,T,99 1040 DATA 13,.5,13,T,10,T,99

Listagem 2

S CLEAR 30440
10 BORDER 0: INK 3: PAPER 0
15 CLS: OVER 1
20 PLOT 128,0
25 DRAW 0,165,2771*PI
30 PLOT 128,0
35 DRAW 0,165,-2771*PI
40 OVER 0: INK 7: BRIGHT 1
45 PRINT AT 0,2; ""COSMAZOIGS"
Adp. CLUBE Z80"
50 PRINT AT 11,11; FLASH 1; "Um
momento!"
55 GO TO 500
90 DATA "7F582488E73838E786240

 0 LET Y = 4095 * VAL A\$ (1) LET Y = Y + 256 * VAL A\$ (2) LET Y = Y + 16 * VAL A\$ (3) LET Y = Y + VAL A\$ (4) IF Y < 30440 THEN GO TO 900 LET A\$ = A\$ (5 TO) LET X = 16 * VAL A\$ (1) + VAL A\$ (2) 520 540 550 560 POKE Y,X: LET Y=Y+1 LET A\$=A\$(3 TO) IF A\$<>T\$ THEN GO TO 660 NEXT L 570 580 700 710720730740 CLS PRINT AT 10,14; "PODE"; AT 11 'GRAVAR" CIRCLE 128,87,30 STOP 300 CLS 910 PRINT AT 10,7; FLASH 1; "ERR NA LINHA"; L 920 FOR H=0 TO 2: BEEP .05,H 930 BORDER H: NEXT H 940 GO TO 920

PUZZZE DE PALAVRAS

SPE 16/48 K

Autor: PAULO CASTELO PORTO

Retirado do "YOUR COMPUTER" Jan/84, este programa funciona tal como os puzzles, que vês publicados em várias

Dar entrada de 10 palavras chave que serão baralhadas num quadro em posições diversas: diagonal, vertical e horizontal. Nota que as palavras podem estar pela ordem inversa. Tens ainda um MENU com 5 opções.

A — Dá-te a resolução

C — Copia p/ a impressora

Q - Para

R — Recomeça

S — Baralha para um novo jogo

ORDER 0: INK 7: BRIG ORDER 0: INK 7: BRIG AURAS ": FOR a=1 TO 31: T 0,0; a= (32-a TO): BEEP NEXT a: LET A=" T 1,0; a= (32-a TO): BEEP NEXT a: LET A=" T 1,0; a= (32-a TO): BEEP INK 7: BRIGHT 1: B GERADOR DE PAL TO 31: PRINT A : BEEP .01,0: ADAPTADO PRINT LET ASS T 3,0; A\$ (32-A TO); BEEP .01,0:

40 FOR A=1 TO 1000: NEXT A: CL PAPER 0; TAB INK 5; 90 PRINT : PRINT TAB 8; "qualqu T tecla"
100 IF INKEYS="" THEN GO TO 100
110 CLS
120 PRINT AT 20,0; "tecla enter. INKEYS "" THEN GO TO 100 130 DIM at (10,10) 140 FOR a=1 TO 10 150 INPUT "palavra t: CL5 : IF LEN bs> "10 letras no maxi MAXIMO! GO TO 50 GO TO 150 NEXT A: CLS PAPER 1; "UM M "" THEN LET as(a) = bs: PRINT AT 7,9; 180 190 DIM gs (20,20): DIM c (20,20)
DIM x (10): DIM y (10)
200 FOR a=1 TO 10
210 FOR b=1 TO 10: IF a=(a,b TO
b) <>" THEN NEXT b
220 LET B=B-1
230 LET x (A) = INT (RND + 20) + 1: LE
y (A) = INT (RND + 20) + 1
240 LET d=="" omento"
190 DIM T (RND+20)+1 250 LET d==d\$+("1" AND yy+b(21) +("2" AND yy+b(21 AND xx)b)+("3" AND xx)b)+("4" AND xx)b AND yy> b)+("5" AND yy>b)+("6" AND yy>b AND xx+b(21)+("7" AND xx+b(21)+("6" AND xx+b(21 AND yy+b(21) 260 LET d=UAL (d\$(INT (RND*(LEN d\$)+1))) di)+1))
270 FOR c=1 TO b: IF qs(xx,yy)=
"THEN LET xx=xx-(d>1 AND d(5)
+(d>5 AND d(9): LET yy=yy-(d>3 AND d(7)+(d<3 OR d=6): NEXT c: GO
TO 290 280 GO TO 230 290 LET ink=INT (RND*4)+4: LET xx=x(a): LET yy=y(a): FOR c=1 TO

ROTINA EM CÓDIGO MÁQUINA

(Resposta à pergunta de Mário Rebelo)

A Pergunta (publicada em Maio, pág. 7):

LDE,N
LDD,N
LDA,("AT"(22))N
RST16
LDA,D
RST16
LDA,E
RST16
LDA,"INR"(1E)N
RST16
LDA,"INR"(1E)N
RST16
LDA,N (car)
RST16
LDA,N (símbolo)
RST16

RET

"A rotina que listo ao lado, como podem ver, coloca em certa
posição do écran um determinado símbolo com a côr que se
pretende. O problema é que,
após o simbolo, surge sempre
um número que varia de linha
para linha. Se souberem,
gostaria que me explicassem
porque é que isso acontece e
como é possível eliminar tal
número do écran".

MÁRIO REBELO/Coimbra

FERNADO PRECES Responde:

"Quando se aplica o caracter 22 como comando "AT", numa rotina em C.M., sem que primeiro se abra o canal

"S" (livre trânsito par o ecran), podem acontecer fenómenos muito estranhos.

É importante frisar que 2 computadores do mesmo modelo, ao depararem com uma instrução imprecisa, momentaneamente podem seguir caminhos diferentes. Assim no meu SPECTRUM 48K introduzi, na íntegra, a sua rotina máquina e nenhum símbolo adicional apareceu no écran. Em contrapartida, escreve o caracter sempre na linha 22, seja qual fôr o número de linha que eu proponha.

Para abrir o canal "S", tem dois procedimentos possíveis:

- a) Programa em Basic, com pequenas rotinas em C.M..
 Quando utilizar a sua rotina, escreva:
 Linha n.º PRINT AT 0,0: RANDOMIZE USR X
- b) Programa em C.M. utilize o seguinte Assembler: LD A,2 CALL 5633 Para abertura do canal "S"

LD B,24 CALL 3652	Se pretender limpar o primeiro o ecran
LD A,22	
RST 16	
LD A,N	
RST 16	Posição PRINT no ecran
L,D A,N	
RST 16	
LD A,16	
RST 16	
	Cor "INK"
LD A,N RST 16	
LD A,17	
RST 16	Cor "PAPER"
LD A,N	
RST 16	
LD A,N	
RST 16	imprime o símbolo
RET	

Para finalizar, devo acrescentar que esta rotina é muito interessante para estudo, mas tem pouca aplicação prática.

Se descodificar alguns programas comerciais existentes no mercado, encontrará rotinas para o efeito, bem mais rápidas e eficazes.

FERNANDO PRECES/Sacavém

SCROLL, BREAK, CONTINUE e VERIFY

RUI CARVALHO/Barreiro expõe "algumas anomalias do teclado do **Spectrum**, que gostaria de ver publicadas, e perguntas que gostaria de ver respondidas":

- Descobri que primindo CAPS SHIFT 3 quando nos é pedido "SCROLL?", este é executado duas vezes sem no-lo pedir — isto pode ser útil especialmente em listagens muito longas.
 - A revista ZX COMPUTING (Ab/Maio 84) apresenta uma nova maneira de fazer "BREAK" num programa:

Primir CAPS SHIFT, SYMBOL SHIFT e 2 ao .mesmo tempo. No entanto, isto nem sempre funcionará. Por exemplo, quando aparece "SCROLL?", caso se prima estas 3 teclas ou outras, nada acontecerá; primindo CAPS SHIFT e SYMBOL SHIFT ao mesmo tempo, aparecerão várias coisas que não nos interessam pois a sua aplicação prática é nula.

Note-se que se se quiser parar o "SCROLL?" a seguir a primir CAPS SHIFT, SYMBOL SHIFT e outra tecla, o ecran subirá (sendo portanto executado o "SCROLL?"). Mas agora, só BREAK parará o "SCROLL?".

Existem também outros casos em que esta combinação não funciona — p. ex.: PAUSE 0 (zero).

 Quando a "SCROLL?" respondemos N, seguido de CONT, porque é que o computador repete ou continua o comando CONTINUE? Porque é que "LIST", no mesmo caso, dá 0 (zero) OR? Por estranho que pareça, VERIFY não pode entrar na constituição de um programa, pois dá erro R: tape loading error (quando o gravador está a funcionar). Existe alguma maneira de VERIFY ser aceite pelo computador?".

RUI CARVALHO espera que os sócios discutam ou esclareçam estas questões. Escrevam-nas ao CLUBE Z80 para publicação.

VERIFIQUE A CAPACIDADE DO SEU ZX SPECTRUM

Publicamos no mês passado (n.º 21, pág. 6) uma indicação para verificar a capacidade do SPECTRUM: PRINT PEEK 23733 ENTER (se 255-48K; se 127-16K).

CARLOS OLIVEIRA/Portimão escreveu-nos o seguinte:

"Julgo ser mais agradável ver no ecran da TV 16 ou 48. Por isso, sugiro que se utilize a instrução PRINT PEEK 23733/4--15.75 ENTER".

Sobre a Rubrica "INTRODUÇÃO À LINGUAGEM MÁQUINA"

JORGE LANDECK/Seia sugere o seguinte:

"Tal como eu, penso que também outros sócios não tiveram a oportunidade de se tornarem tal, senão recentemente. Assim, alguns de nós apenas possuímos as revistas mais recentes o que, constitui, pelo menos para mim, uma grande lacuna (...). Venho pois sugerir que se faça, por exemplo a edição de uma revista especial, onde se resumisse a rubrica "Introdução à Linguagem Máquina" que penso de especial importância".

Que pensam os outros sócios desta ideia? Escrevam-nos.

TROCO PROGRAMAS PARA O SPECTRUM



CONTACTAR: TIAGO RAMALHO

R. CLEMENTE MENÉRES, 47-3.° D.

4000 PORTO

DUAS "PEQUENAS" ROTINAS

SPECTRUM

BEEP NO SPECTRUM

Talvez muitos dos possuidores do Spectrum estejam desapontados com o comando BEEP.

Aqui apresentamos uma pequena rotina em BASIC que lhe dará novas perspectivas sobre este comando:

```
10 READ b: RESTORE
20 FOR i=1 TO 32
30 READ a
40 BEEP 0.2,a: BEEP .05,a
50 LET i
70 DATA 13,11,13,9,4,9,1,1
60 DATA 13,11,13,9,4,9,1,1
90 DATA 13,15,16,15,16,13,15,1
```

NÃO PERMITIR O MERGE

Qualquer programa em BASIC pode ser parado, usando a instrucção MERGE. Para evitar a leitura do seu programa, apresentamos a seguir duas linhas que devem ser adicionadas à listagem e aproveitadas para a gravação do mesmo (GOTO 9998).

Desta forma será impossível fazer MERGE ou BREAK, ficando o programa a entrar com LOAD""CODE.

```
9998 POKE 23613,82: POKE 23614,6
4: SAVE "(nome do programa)" COD
E 23552, (valor aprox.) +1000

9999 GO TO (inicio do programa)

Para achar o comprimento aprox.
Faca: PRINT 41472-(65535-USR 796
```

TOPS EM INGLATERRA

OS MICROS E PROGRAMAS (Spectrum) MAIS VENDIDOS, EM INGLATERRA, NA PRIMEIRA SEMANA DE JULHO (informações da revista PERSONAL COMPUTER NEWS, 14 Julho, 1984, n.º 69)

MICROS

Preço até £: 1 000														
1 - Spectrum													£	99
2 - CBM 64													£	199
3 - Electron					 *		. ,						£	199
4 - Vic 20													£	145
5 - BBC B													£	399
6 - Oric Atmos										. ,			£	175
7 - Memotech 500													£	275
8 - Atari 800XL													£	250
9 - ORIC													£	99
10 - Dragom													£	150

Preço superior £: 1 000
1 - IBMPC £ 2 390
2 - Apricot £ 1 760
3 - Apple III £ 2 755
4 - Sirius £ 2 525
5 - TS 1603 £ 2 640
6 - DEC Rainbow £ 2 359
7 - Compaq £ 1 960
8 - Wang Professional £ 3 076
9 - Philips P2000 C £ 1 484
10 - LSI Octópus £ 1 760

PROGRAMAS SPECTRUM (Jogos)

1 - SABRE WULF*	11 - CODE NAME MAT
2 — TLL	12 — VALHALLA*
3 — PSYTRON	13 - NIGHT GUNNER*
4 — MUGSY	14 — CAVELON
5 - FIGHTER PILOT*	15 - SCUBA DIVE*
6 — HULK	16 - CHUKKIE EGG*
7 — JACK & B'STALK	17 - ATIC ATAC*
8 - LORDS OF MIDNIGHT	18 — ANTICS
9 — TRASHMAN*	19 - FOOTBALL MANAGER*
10 - JET SET WILLY*	20 - CHEQUERED FLAG*

OS 10 MAIS VENDIDOS NO CLUBE Z80

(JOGOS SPECTRUM)

- 1-JET SET WILLY
- 2-FIGHTER PILOT
- 3 PHEENIX
- 4 CHEQUERED FLAG
- 5 NIGHT GUNNER
- 6 HUNTER KILLER
- 7 SIMULADOR DE VOO (Psion)
- 8 SPACE SHUTTLE
- 9 PINBALL
- 10 ATIC ATACK

400\$00

NOVOS PROGRAMAS

SPECTRUM

JOGOS	Preço
 BLACK PLANET (48K) — Séc. XXI: Black Planet, o planeta pirata serve de base para elementos criminosos. Você é "Starmagon" de uma patrulha galática que tem como objectivo a destruição das vias espaciais desses elementos. Para isso, terá de conseguir as 7 peças-chave necessárias à destruição do BLACK PLANET. 	400\$00
 FRED (48K) — O herói do jogo explora um labirinto cheio de monstros, com o objectivo de apanhar tesouros, dispondo de uma arma com 6 balas. 	400\$00
 GRID RUN (16/48K) — Um labirinto com 2 automóveis. Sendo condutor de um, percorra rapidamente todo o labirinto, evitando o seu perdeguidor. MUNCHER (16/48K) — Outra versão de "Pacman" com 9 níveis de dificuldade e possibilidade de um ou 2 jogadores. 	400 \$ 00 400 \$ 00
 PHEENIX (16/48K) — Jogo de arcádia muito popular, com vários níveis de dificuldade e diferentes fases durante a invasão inimiga. O jogo termina com a destruição da nave-mãe que é protegida ainda por alguns invasores. 	400\$00
 POGO (48K) — Percorra uma pirâmide de hexágonos que vão mudando de côr, aquando da sua passagem. Há perseguidores a impedir o êxito da sua missão. 	400\$00
 TANKS (16/48K) — O seu objectivo é duplo: Destuir os tanques inimigos sem os deixar avançar além da linha de fogo e procurar uma base para abastecimento de combustível e munições. Atenção! O caminho está minado. Por isso oriente-se pelas diferentes cores correspondentes aos vários detectores de minas. 	400\$00
 TRASHMAN (48K) — A finalidade do jogo é apanhar o número de contentores do lixo indicado, transportando-os até ao carro e colocando-os no seu lugar. 	400\$00
• UBOAT HUNT (48K) — Versão traduzida do "Hunter Killer" (destruir submarinos inimigos).	400\$00
• ZOMBIES (16/48K) — Jogo de arcádia do género "Invadens", com apresentação gráfica bastante variada e com diferentes	

DESCONTO DE 20% PARA SÓCIOS DO CLUBE-VENDAS NA SEDE OU À COBRANÇA

OS PROGRAMAS SEGURANÇA E CDU

fases aumentando o nível de dificuldade.

Nos **novos programas** anunciados em Maio, publicámos SEGURANÇA E CDU que, apesar da nossa insistência com o seu editor/autor, ainda não nos foram remetidos.

Lamentamos este facto e apenas podemos pedir desculpa a todos os sócios que os requisitaram. Os seus pedidos continuam "na gaveta" e serão atendidos quando dispusermos dos referidos programas.

